Amendments to the claims:

The listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of the Claims:

1. (Currently Amended) In a Java computing environment, a method of customizing a Java runtime [[-]] environment for a Java class file which is executed by a virtual machine said method comprising:

reading a java class file, which includes a plurality of Bytecodes, prior to loading or executing the class file by the virtual machine;

analyzing said java class file, for one or more runtime[[-]]attributes associated with runtime performance of the java class file, prior to loading or executing the class file by the virtual machine;

marking one or more Bytecodes of said class file, based on said analyzing of said class file, prior to loading or executing the class file by the virtual machine;

generating at least one ~~run-time~~ runtime attribute for each one of said one or more marked Java Bytecodes prior to loading or executing the class file by the virtual machine;

reading by the virtual machine, during the load time of the class file into the virtual machine, the at least one ~~run-time~~ runtime attribute for each one of said one or more marked Java Bytecodes;

loading by the virtual machine at least one runtime feature associated with the at least one ~~run-time~~ runtime attribute prior to execution of said class file by the virtual machine; and

executing said class file by said virtual machine in a runtime environment that includes the at least one runtime feature associated with the at least one ~~run-time~~ runtime attribute, thereby allowing the runtime environment to be customized based on the analyzing, marking, and generating of the at least one ~~run-time~~ runtime attribute.

2. (Original) A method as recited in claim 1, wherein said loading operates to load a feature only if said feature has an associated attribute.

3. (Original) A method as recited in claim 1, wherein said marking is performed by a Java compiler extension.

4. (Previously Presented) A method as recited in claim 1, wherein said marking is performed by a software tool suitable for analyzing runtime performance of a Java application associated with said java class file in the runtime environment.

5. (Original) A method as recited in claim 1, wherein said one or more marked Java Bytecodes are associated with a Java method.

6. (Original) A method as recited in claim 5, wherein said one or more marked Java Bytecodes is associated with a Java object that has an attribute that is of interest.

7. (Original) A method as recited in claim 6, wherein said attribute is the life span, size, or class of said Java object.

8. (Original) A method as recited in claim 7, wherein said generating of said at least one attribute is performed by a software module that operates to generate said at least one attribute in an attributes table of said class file as the last attribute.

9-10. (Cancelled)

11. (Currently Amended) A Java computing environment: a virtual machine for executing a Java class file, wherein the virtual machine is capable of:

reading a java class file which includes a plurality of Bytecodes prior to loading or executing the class file by the virtual machine;

analyzing said java class file, for one or more runtime [[-]] attributes associated with runtime performance of the java class file, prior to loading or executing the class file by the virtual machine;

marking one or more Bytecodes of said class file, based on said analyzing of said class file, prior to loading or executing the class file by the virtual machine;

generating at least one run-time runtime attribute for each one of said one or more marked Java Bytecodes prior to loading or executing the class file by the virtual machine;

reading by the virtual machine, during the load time of the class file into the virtual machine, the at least one ~~run-time~~ <u>runtime</u> attribute for each one of said one or more marked Java Bytecodes;

loading by the virtual machine at least one runtime feature associated with the at least one ~~run-time~~ <u>runtime</u> attribute prior to execution of said class file by the virtual machine; and

executing said class file by said virtual machine in a runtime environment that includes the at least one runtime feature associated with the at least one ~~run-time~~ <u>runtime</u> attribute, thereby allowing the runtime environment to be customized based on the analyzing, marking, and generating of the at least one ~~run-time~~ <u>runtime</u> attribute.

12. (Cancelled)

13. (Previously Presented) A Java computing environment as recited in claim 11, wherein a software module operates as a runtime performance manager and operates to ensures that said at least one feature is appropriately loaded into said virtual machine.

14. (Original) A Java computing environment as recited in claim 13, wherein said runtime performance manager includes a database that can be used as input by the second software module to generate said at least one attribute.

15. (Previously Presented) A Java computing environment as recited in claim 11, wherein a compiler extension or a software tool is utilized for analyzing said class file.

16. (Currently Amended) A computer readable media including computer program code for customizing a Java runtime environment for a Java class file which is executed by a virtual machine, said computer readable media comprising:

computer program code for reading a java class file which includes a plurality of Bytecodes prior to loading or executing the class file by the virtual machine;

computer program code for analyzing said java class file, for one or more runtime [[-]] attributes associated with runtime performance of the java class file, prior to loading or executing the class file by the virtual machine;

SUN1P835/P6076   4 of 8

computer program code for marking one or more Bytecodes of said class file, based on said analyzing of said class file, prior to loading or executing the class file by the virtual machine;

computer program code for loading at least one ~~run-time~~ <u>runtime</u> attribute for each one of said one or more marked Java Bytecodes prior to loading or executing the class file by the virtual machine;

computer program code for providing by the virtual machine at least one runtime feature associated with the at least one ~~run-time~~ <u>runtime</u> attribute prior to execution of said class file by the virtual machine; and

computer program code for executing said class file by said virtual machine in a runtime environment that includes the at least one runtime feature associated with the at least one ~~run-time~~ <u>runtime</u> attribute, thereby allowing the runtime environment to be customized based on the analyzing, marking, and generating of the at least one ~~run-time~~ <u>runtime</u> attribute.

17. (Original) A computer readable media as recited in claim 16, wherein said computer program code for loading operates to load a feature only if said feature has an associated attribute.

18. (Original) A computer readable media as recited in claim 16, wherein said computer program code for marking is performed by a Java compiler extension.

19. (Original) A computer readable media as recited in claim 16, wherein said computer program code for marking is performed by a software tool suitable for analyzing performance of a Java application in the runtime environment.

20. (Previously Presented) A computer readable media as recited in claim 19, wherein said marked Java Bytecode is associated with a Java object that has an attribute that is of interest.

21. (Previously Presented) A computer readable media as recited in claim 20, wherein said attribute is the life span, size, or class of said Java object.

SUN1P835/P6076   5 of 8